

Levis, Stalin, and Software

“Real world” is turning into one of those hackneyed terms in software development. User interface design is obsessed with finding real world metaphors that work in applications. Similarly, object-oriented design aims to create software objects that model their real world counterparts. As you can see, we as developers are captivated with bringing real world *content* into our software products — but how often do we look at software systems themselves and see how they are related to other systems in our world? In that light, let’s take a look at four lessons that can be gleaned from “human systems.” I believe that by looking at these truths, we can learn to design better software.

Rugged Individualism. *Successful systems let an individual be himself or herself.* In the retail world, companies like Levi Strauss are now using “mass customization” to provide better service to its customers — in this case, a custom-fit pair of jeans. In the political world, the history of the former Soviet Union shows us it is impossible to truly advance and prosper without a democratic society. This same need for freedom also exists in the software realm. While people worked with inflexible, controlled programs for years, they typically did their work in spite of them, rarely flourishing in such an environment. (In fact, as was the case with the thriving Soviet black market, much of the energy may be spent going *around* a rigid system, not working within it.) In other words, your users need freedom to “do their own thing.” Alan Cooper writes in his thought-provoking book *About Face: The Essentials of User Interface Design* [IDG Books, 1995], “Users really like personalization. It allows them to feel part of the computing process; to buy into the task being performed.” Windows 95 — which has an interface geared towards customization — provides a great example of this democratic ideal. Don’t design your software as Stalin would; if so, it will never produce the effects for which the software was designed.

Balanced Approach. *Paradoxically, successful systems also provide structure.* Too much individualism can be detrimental, and can leave people feeling lost. As a father, I have discovered the intrinsic need that children have for

structure and discipline — not unadulterated freedom. Bringing this discussion back to our software world, we can say that personalization is important, but only within a strong framework. Moreover, within this backdrop, the individual needs of users are quite different. For example, some users will change their Windows color schemes daily, while others will never stray from the default settings. Your applications must account for both types of users. Cooper concludes: “Personalization is one of those idiosyncratically modal things. People either like it or they don’t.”

Decentralization. *Successful systems are decentralized.* Maintaining a central focus always sounds great in principle, especially to those in leadership. But the reality is that “centralized” systems rarely work. Think of the economic world in the twentieth century: free economies prospered while centrally planned economies of communist governments floundered. This also holds true in our software domain. Server-based applications running on terminal emulators are relics. While the World Wide Web renews the call by some for centralized solutions, the fact is that the Web will be successful because there is “freedom” on the desktop, not because people will rush out to buy Internet terminals. People can accept centralization when it makes sense, but not when it limits their individualism.

Fallen Systems. *Successful systems are built with realistic expectations.* As much as we hate to admit it, human

systems are “fallen systems.” Realizing this, our Founding Fathers structured a divided government in such a way as to prevent a single individual from achieving absolute power. Their premise was to plan for the possibility of problems. However, the temptation in system design is often the opposite: to build a perfect world. People have always tried to achieve this — be it through government, economics, education, or philosophy — but it has never been achieved. So too in the software realm. The expectations of developers and users in software are so high that we are bound to be disillusioned when problems develop. While we should not excuse bugs, we do need to put them into the proper context. Software is a human invention, and just like other products of the human intellect, it is fallible. If we recognize this reality, we can plan for problems and put appropriate safeguards into place.

Not in a Vacuum. Software development is never performed in a vacuum. The products we develop are used by people living in the “real world.” Thus, the same way people react to political, economic, and commercial systems will be the way users respond to your software. ▲

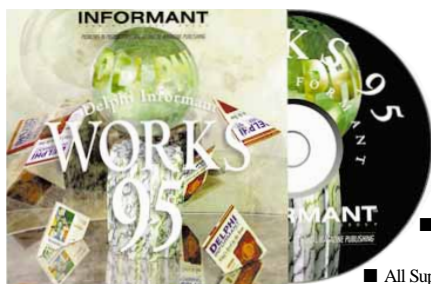
— Richard Wagner

Richard Wagner is Contributing Editor to Delphi Informant and the Chief Technology Officer of Acadia Software in the Boston, MA area. He welcomes your comments at rwagner@acadians.com.



The Must Have Reference Source For The Serious Delphi™ Developer

The Entire Text of Over 60 Technical Articles Appearing in
Delphi™ Informant® in 1995



The Delphi™ Informant® Works 1995
CD-ROM Includes:

- Over 60 Technical Articles
- Text and Keyword Search Capability

- Improved Speed and Performance

- All Supporting Code and Sample Files

- 16-Page Color Booklet

- Electronic Version of *Delphi Power Tools* Catalog

- Electronic Version of *Borland C++ Power Tools* Catalog

- Third-Party Add-In Product Demos

- CompuServe Starter Kit with \$15 Usage Credit.

A \$105 Value Available Now for only

\$39.95

California residents add 7¹/₄% Sales Tax,
plus \$5 shipping & handling for US orders.
(International orders add \$15 shipping & handling)

Call Now Toll Free 1-800-88-INFORM

1-800-884-6367 Ask for offer # PI95 To order by mail,
send check or Money Order to:

Informant Communications Group, Inc.

ATTN: Works CD offer # PI95

10519 E. Stockton Blvd, Suite 142 Elk Grove, CA 95624-9704

or Fax your order to

916-686-8497

Get Informed!

Subscribe to Delphi Informant, The
Complete Monthly Guide to Delphi
Development and stay ahead of the rapid
application development curve.

Order Now and Get One Issue FREE!

For a limited time you can receive the first issue FREE plus 12 additional
issues for only \$49.95 That's nearly 25% off the yearly cover price!



Each big issue of *Delphi Informant* is packed with
Delphi tips, techniques ,
news, and more!

- Client/Server Application Development

- Delphi Programming

- Using the Borland Database Engine

- Object-Oriented Programming

- Creating Reusable Components

- Product Reviews

- News from the Delphi Community

- Delphi User Group Information

Payment Method...

- ☐ Check (payable to Informant Communications Group)

- ☐ Purchase Order--
Provide Number _____

- ☐ Visa

- ☐ Mastercard

- ☐ American Express

Card Number _____

Expiration Date _____

Signature _____

International rates

Magazine-only

\$54.95/year to Canada

\$74.95/year to Mexico

\$79.95/year to all other countries

Magazine AND Companion Disk Subscriptions

\$124.95/year to Canada

\$154.95/year to Mexico

\$179.95 to all other countries

California Residents add 7¹/₄% sales tax
on disk subscription

☐ **YES**

I want to sharpen my Delphi Programming skills. I've checked the subscription plan I'm
interested in below

☐ **Magazine-Only Subscription Plan...**

13 Issues Including One Bonus Issue at \$49.95.

☐ **Magazine AND Companion Disk Subscription Plan...**

13 Issues and Disks Including One Bonus Issue and Disk at \$119.95

The Delphi Informant Companion Disk contains source code, support files, examples, utilities, samples, and more!

☐ **Delphi Informant Works 1995 CD-ROM \$39.95**

US residents add \$5 shipping and handling. International customers add \$15 shipping and handling.

Name _____

Company _____

Address _____

City _____ State _____ Zip Code _____

Country _____ Phone _____

FAX _____ e-Mail _____

To order, mail or fax the form below or call
(916) 686-6610 Fax: (916) 686-8497